This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method of accessing a data structure, the method comprising:

initializing a flux count associated with a data structure to an even value;

in response to a request to modify the data structure, sequentially and in order:

incrementing the flux count to an odd value;

acquiring an exclusive serialization mechanism for the data structure;

modifying the data structure;

releasing the exclusive serialization mechanism; and

incrementing the flux count to an even value; and

in response to a request to access data from the data structure, sequentially and in order:

copying the flux count to obtain a <u>first</u> copy of the flux count;

copying the requested data from the data structure to obtain a copy of the requested data;

<u>copying the flux count to obtain a second copy of the flux count;</u> and

determining that the copy of the requested data is valid if the copy of the flux count is an even value and the <u>first and second copies</u> ~~copy~~ of the flux count <u>are</u> ~~is still~~ equal to the flux count ~~after the copy of the requested data is obtained~~.

2. (Currently Amended) The method of claim 1, further comprising, in response to the request to access data from the data structure determining that the copy of the requested data is not valid if either the copy of the flux count is an odd value or the ~~copy~~

first and second copies of the flux count ~~is~~ <u>are</u> not equal ~~to the flux count after the copy of the requested data is obtained~~.

3. (Original) The method of claim 2, further comprising, in response to determining that the copy of the requested data is not valid, acquiring a shared serialization mechanism for the data structure and obtaining a copy of the requested data while the shared serialization mechanism is acquired.

4. (Original) The method of claim 2, further comprising, in response to determining that the copy of the requested data is not valid, repeating the copying of the flux count and the copying of the requested data from the data structure until the copy of the flux count is an even value and the copy of the flux count is still equal to the flux count.

5. (Currently Amended) A computer-implemented method of accessing a data structure, the method comprising:

in connection with modifying the data structure:

prior to modifying the data structure, updating a flux indicator associated with the data structure from a first state to a second state to indicate that the data structure is in the process of being modified; and

after modifying the data structure, updating the flux indicator to a third state to indicate that the data structure is no longer in the process of being modified, wherein the third state is different from each of the first and second states; and

in connection with accessing data from the data structure:

obtaining a first copy of the flux indicator in connection with obtaining a copy of data from the data structure;

obtaining a second copy of the flux indicator after obtaining the copy of the data from the data structure; and

determining that the copy of the data from the data structure is valid if the first copy of the flux indicator does not indicate that the data structure is in the process of being modified and if the first and second copies of the flux indicator have the same state; wherein the flux indicator includes a count value, wherein the first state of the flux indicator includes an even count value, wherein updating the flux indicator from the first state to the second state includes incrementing the flux indicator to an odd count value, wherein updating the flux indicator to the third state includes incrementing the flux indicator to an even count value, and wherein determining that the copy of the data from the data structure is valid includes determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an even count value.

6.- 13. (Canceled).

14. (Original) The method of claim 5, wherein accessing the data from the data structure is performed without acquiring a serialization mechanism.

15. (Original) The method of claim 5, wherein accessing the data from the data structure further comprises accessing the data from the data structure after acquiring a shared serialization mechanism in response to determining that the copy of the data from the data structure is not valid.

16. (Previously Presented) The method of claim 5, wherein accessing the data from the data structure further comprises determining that the copy of the data from the data structure is not valid, and in response thereto, repeatedly obtaining of the first copy of the flux indicator, obtaining the copy of data from the data structure, and obtaining the second copy of the flux indicator until the first copy of the flux indicator does not indicate that the data structure is in the process s of being modified and the first and second copies of the flux indicator have the same state.

17. (Original) The method of claim 5, wherein the data from the data structure includes a plurality of fields, wherein obtaining the first copy of the flux indicator in connection with obtaining the copy of the data from the data structure includes obtaining the first copy of the flux indicator in connection with obtaining copies of the plurality of fields, and wherein determining that the copy of the data from the data structure is valid includes determining that the copies of the plurality of fields are valid based upon the flux indicator.

18. (Original) The method of claim 5, wherein modifying the data structure further includes, after updating the flux indicator from the first state to the second state and before updating the flux indicator to the third state, acquiring an exclusive serialization mechanism for the data structure, thereafter modifying the data structure, a thereafter releasing the exclusive serialization mechanism.

19. (Original) The method of claim 5, wherein the data structure comprises a journaled object, and wherein the data in the data structure includes an indication of whether the journaled object is in a standby mode.

20. (Currently Amended) An apparatus, comprising:

a memory and at least one processor;

a data structure resident in the memory;

a flux indicator associated with the data structure;

first program code configured to execute on the at least one processor to modify the data structure, the first program code configured to, prior to modifying the data structure, update the flux indicator from a first state to a second state to indicate that the data structure is in the process of being modified, and, after modifying the data structure, update the flux indicator to a third state to indicate that the data structure is no longer in the process of being modified, wherein the third state is different from each of the first and second states; and

second program code configured to execute on the at least one processor to access data from the data structure, the second program code configured to obtain a first copy of the flux indicator in connection with obtaining a copy of data from the data structure, obtain a second copy of the flux indicator after obtaining the copy of the data from the data structure, and determine that the copy of the data from the data structure is valid if the first copy of the flux indicator does not indicate that the data structure is in the process of being modified and if the first and second copies of the flux indicator have the same state;

wherein the flux indicator includes a count value, wherein the first state of the flux indicator includes one of an even and an odd count value, wherein the first program code is configured to, if the first state of the flux indicator is an even count value, update the flux indicator from the first state to the second state by incrementing the flux indicator to an odd count value, and if the first state of the flux indicator is an odd count value, update the flux indicator from the first state to the second state by incrementing the flux indicator to an even count value, wherein the first program code is configured to, if the first state of the flux indicator is an even count value, update the flux indicator to the third state by incrementing the flux indicator to an even count value, and if the first state of the flux indicator is an odd count value, update the flux indicator to the third state by incrementing the flux indicator to an odd count value, and wherein the second program code is configured to, if the first state of the flux indicator is an even count value, determine that the copy of the data from the data structure is valid by determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an even count value, and if the first state of the flux indicator is an odd count value, determine that the copy of the data from the data structure is valid by determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an odd count value.

21.-24. (Canceled).

25. (Original) The apparatus of claim 20, wherein the second program code is configured to access the data from the data structure without acquiring a serialization mechanism.

26. (Original) The apparatus of claim 20, wherein the second program code is configured to access the data from the data structure by accessing the data from the data structure after acquiring a shared serialization mechanism in response to determining that the copy of the data from the data structure is not valid.

27. (Previously Presented) The apparatus of claim 20, wherein the second program code is configured to access the data from the data structure by determining that the copy of the data from the data structure is not valid, and in response thereto, repeatedly obtaining of the first copy of the flux indicator, obtaining the copy of data from the data structure, and obtaining the second copy of the flux indicator until the first copy of the flux indicator does not indicate that the data structure is in the process of being modified and the first and second copies of the flux indicator have the same state.

28. (Original) The apparatus of claim 20, wherein the data from the data structure includes a plurality of fields, wherein the second program code is configured to obtain the first copy of the flux indicator in connection with obtaining the copy of the data from the data structure by obtaining the first copy of the flux indicator in connection with obtaining copies of the plurality of fields, and wherein the second program code is configured to determine that the copy of the data from the data structure is valid by determining that the copies of the plurality of fields are valid based upon the flux indicator.

29. (Original) The apparatus of claim 20, wherein the first program code is configured to modify the data structure by, after updating the flux indicator from the first state to the second state and before updating the flux indicator to the third state, acquiring

an exclusive serialization mechanism for the data structure, thereafter modifying the data structure, a thereafter releasing the exclusive serialization mechanism.

30. (Original) The apparatus of claim 20, wherein the data structure comprises a journaled object, and wherein the data in the data structure includes an indication of whether the journaled object is in a standby mode.

31. (Currently Amended) A program product, comprising:

first program code configured to modify a data structure, the first program code configured to, prior to modifying the data structure, update a flux indicator associated with the data structure from a first state to a second state to indicate that the data structure is in the process of being modified, and, after modifying the data structure, update the flux indicator to a third state to indicate that the data structure is no longer in the process of being modified, wherein the third state is different from each of the first and second states;

second program code configured to access data from the data structure, the second program code configured to obtain a first copy of the flux indicator in connection with obtaining a copy of data from the data structure, obtain a second copy of the flux indicator after obtaining the copy of the data from the data structure, and determine that the copy of the data from the data structure is valid if the first copy of the flux indicator does not indicate that the data structure is in the process of being modified and if the first and second copies of the flux indicator have the same state; and

a tangible computer readable storage medium bearing the first and second program code; wherein the flux indicator includes a count value, wherein the first state of the flux indicator includes one of an even and an odd count value, wherein the first program code is configured to, if the first state of the flux indicator is an even count value, update the flux indicator from the first state to the second state by incrementing the flux indicator to an odd count value, and if the first state of the flux indicator is an odd count value, update

the flux indicator from the first state to the second state by incrementing the flux indicator to an even count value, wherein the first program code is configured to, if the first state of the flux indicator is an even count value, update the flux indicator to the third state by incrementing the flux indicator to an even count value, and if the first state of the flux indicator is an odd count value, update the flux indicator to the third state by incrementing the flux indicator to an odd count value, and wherein the second program code is configured to, if the first state of the flux indicator is an even count value, determine that the copy of the data from the data structure is valid by determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an even count value, and if the first state of the flux indicator is an odd count value, determine that the copy of the data from the data structure is valid by determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an odd count value.

32. (Canceled).

33. (New) A computer-implemented method of accessing a data structure, the method comprising:

in connection with modifying the data structure:

prior to modifying the data structure, updating a flux indicator associated with the data structure from a first state to a second state to indicate that the data structure is in the process of being modified; and

after modifying the data structure, updating the flux indicator to a third state to indicate that the data structure is no longer in the process of being modified, wherein the third state is different from each of the first and second states; and

in connection with accessing data from the data structure:

obtaining a first copy of the flux indicator in connection with obtaining a copy of data from the data structure;

obtaining a second copy of the flux indicator after obtaining the copy of the data from the data structure; and

determining that the copy of the data from the data structure is valid if the first copy of the flux indicator does not indicate that the data structure is in the process of being modified and if the first and second copies of the flux indicator have the same state;

wherein the flux indicator includes a count value, wherein the first state of the flux indicator includes an odd count value, wherein updating the flux indicator from the first state to the second state includes incrementing the flux indicator to an even count value, wherein updating the flux indicator to the third state includes incrementing the flux indicator to an odd count value, and wherein determining that the copy of the data from the data structure is valid includes determining if the count values for the first and second copies of the flux indicator are equal and determining if the first copy of the flux indicator is set to an odd count value.